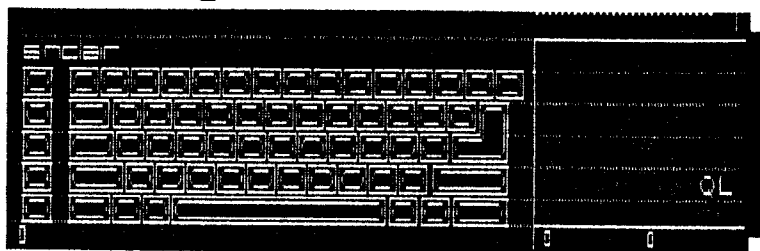


AOUT  
SEPT  
1989

# QL\_DOC

#11

\$1.50



## Le Bulletin Québécois pour les Utilisateurs Sinclair QL

QL\_DOC est publié à tous les 2 mois et est réalisé avec l'aide du logiciel PAGE DESIGNER II avec OMNI-DUMP de SECTOR SOFTWARE. L'impression se fait lentement mais sûrement sur une imprimante STAR NX-1000.

Vos articles, commentaires, suggestions sont appréciés.

Rédacteur: Réal Cagnon, 8286 St-Hubert, Montréal (Québec)  
CANADA H2P 1Z3 (514) 381-6462

Abonnement 1 an (6 nos) : \$12.00

### ... REMark ...

Ce numéro de QL\_DOC contient 3 programmes BASIC qui peuvent être obtenu directement de QL\_DOC, vous sauvant ainsi la tâche fastidieuse de les taper directement au clavier. La façon de procéder est très simple. Envoyer \$1 plus le support sur le quel vous voulez les recevoir. Habituellement, QL\_DOC renvoie par la même occasion des petits extras. Si vous trouvez des "bugs", QL\_DOC aimerait bien le savoir. Par exemple, il paraît que DLO\_BAS ne fonctionnerait pas bien sur les imprimante SEIKOSHA/QL printer, je ne peux pas le confirmer car sur mon imprimante, ça marche numéro 1!!!

Si vous recherchez un QL comme "backup", SHARP vend des QLs à \$99 US, c'est peut-être le temps d'en profiter.

A la prochaine...

# Sinclair News

**SECTOR SOFTWARE: Nouveaux produits.....**

SECTOR SOFTWARE offre une nouvelle ROM pour le QL, elle s'appelle NGUK. Ecrite par Richard Alexander, elle corrige les "BUGs" des anciennes versions JM/JS/MG, quelques nouvelles commandes ont été ajoutées comme par exemple une commande TRACE pour le SuperBASIC. Le prix de vente est 24 livres sterling.

OVERDRIVE est un programme utilitaire pour imprimante qui offre l'équivalent 255 "TRANSLATES" possibles avec une possibilité d'utiliser jusqu'à 80 caractères pour chacune d'elles! OVERDRIVE peut également insérer un graphique en plein milieu d'un texte lors de l'impression de ce dernier à partir de QUILL. Le prix est de 16 livres sterling.

SECTOR SOFTWARE, 39 Wray Crescent, Ulms Walton,  
Leyland Lancs, PR5 3HH, England

**LA FIN DES MICRODRIVES.....**

La compagnie britannique ABLEX, annonce son intention de terminer la fabrication des cartouches µdrives d'ici la fin de l'année 1989. On invoque 2 raisons:

-le marché Qlien pour les µdrives diminue rapidement, en partie à cause de la popularité des disquettes.

-BASF, qui fournit le ruban magnétique, termine la production du ruban utilisé par les cartouches.

**DES NOUVELLES DU SINCLAIR N. AMERICA USER GROUP...**

(Sources: Sinc-Link -Toronto User Group  
SNUG Bytes - Milwaukee User group)

J'ai reçu des nouvelles plus ou moins fraîches à propos du SNUG ( voir QL\_DOC précédent ). C'est le 6 mai dernier qu'a eu lieu une réunion nationale dans le but d'élire un comité exécutif. 35 membres présents ont élu le comité suivant:

Président : Mel Nathanson  
Premier vice-président : Paul Holmgren  
Deuxième vice-président : Basil Wentworth  
Secrétaire : Audrey Curnutt  
Trésorier : John Cushran

Le SNUG s'occupe de tout les ordinateurs appartenant à la famille SINCLAIR, du ZX81 au QL.

**abonnement:** 1 an (6 nos) .....\$12.00  
Réal Gagnon, RL\_DOC éditeur  
8206 St-Hubert Montréal (Qc) H2P 1Z9

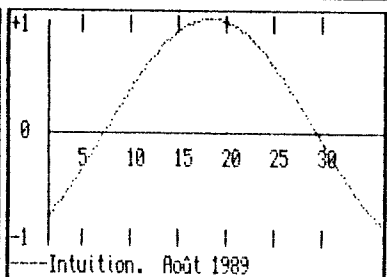
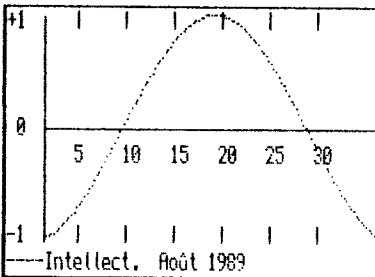
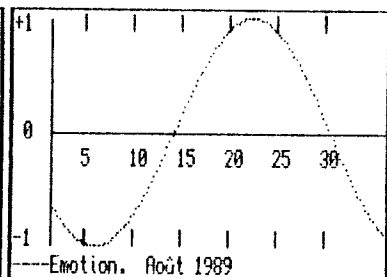
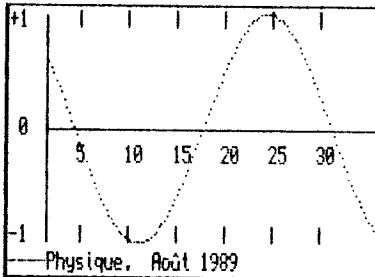
Si vous demandez des renseignements concernant le SNUG  
veuillez inclure 1\$ pour couvrir les frais postaux...

SNUG, 7515 Arbordale Dr., Port Richey, FL 34668, USA

## Biorythmes\_bas

par JC Touzin

Le programme BIORYTHME\_BAS, écrit par notre ami JC Touzin,  
a besoin d'un programme de copie graphique d'écran. Le  
fichier GPRINT\_PRT présent sur le µdrive contenant le  
programme EASEL (celui qui fait des graphiques) peut très  
bien faire l'affaire. Il s'agira d'ajuster la ligne 100 en  
conséquence, pour le charger en mémoire.



Biorythme de Réal Gagnon

Date de naissance: 10 Novembre 1962

Encore (o/n)?

BIORHYTHMES\_BAS par JC Touzin page 1

```

1 REMark BIORHYTHMES_BAS
2 REMark écrit par JC Touzin, La Sarre (Québec)
3 :
99 REMark remplacer "raml_hc_bin" par votre programme de
   copie d'écran, par "mdvl_GPRINT_PRT" de EASEL
   par exemple...
100 CLEAR: a=RESPR(1000): LBYTES raml_hc_bin,a
110 MODE 4
120 OPEN#4,scr: OPEN#5,scr: OPEN#6,scr: OPEN#7,scr
130 REPEAT bio
140 intro
150 saisie
160 jour mois ac,mc
170 courbe
180 INPUT#0,, "Encore (o/n)? ";o_n$
190 IF o_n$="n": EXIT bio
200 END REPEAT bio
210 CLOSE#4: CLOSE#5: CLOSE#6: CLOSE#7:STOP
220 DEFINE PROCEDURE saisie
230 REMark saisie des données
240 PRINT#0,"Programme de calcul du biorythme."
250 INPUT#0,"Copie sur imprimante (o/n)? ";i$
260 INPUT#0,"Nom de la personne ";nom_b$
270 PRINT#0,"Date de naissance:"
280 INPUT#0,"Jour (1-31)? ";jn;" Mois (1-12)? ";mn;
   " Année (ex. 1987)? ";an
290 SELECT ON mn
300 = 1 TO 2: factn=janv_fev(jn,mn,an)
310 = 3 TO 12: factn=autres_mois(jn,mn,an)
320 END SELECT
330 PRINT#0,"Mois et année du calcul:"
340 INPUT#0,"Mois (1-12)? ";mc;" Année (ex. 1986)? ";ac: jc=1
350 SELECT ON mc
360 = 1 TO 2: factc=janv_fev(jc,mc,ac)
370 = 3 TO 12: factc=autres_mois(jc,mc,ac)
380 END SELECT
390 REMark PRINT factn: PRINT factc
400 s=23: t=28: z=33: ii=38
410 END DEFINE saisie
420 DEFINE FUNCTION janv_fev (jour,mois,annee)
430 RETURN 365#annee+jour+31*(mois-1)+INT((annee-1)/4)-
   INT(.75*(INT(((annee-1)/100)+1)))
440 END DEFINE janv_fev
450 DEFINE FUNCTION autres_mois (jour,mois,annee)
460 RETURN 365#annee+jour+31*(mois-1)+INT(.4#mois+2.3)+
   INT(annee/4)-INT(.75*(INT(annee/100)+1))
470 END DEFINE autres_mois
480 DEFINE PROCEDURE jour_mois (annee,mois)
490 SELECT ON mois
500 =2: jour=28

```

BIORYTHMES\_BAS par JC Touzin page II

```

510 =4,6,9,11: jour=30
520 =1,3,5,7,8,10,12: jour=31
530 END SElect
540 IF jour=28 AND annee/4=INT (annee/4): jour=29: END IF
550 SElect ON jour
560 =28: si=139.79
570 =29: si=144.16
580 =30: si=148.58
590 =31: si=153
600 END SElect
610 END DEfine jour_mois
620 DEfine PROCEDURE quel_mois (mois)
630 SElect ON mois
640 =1: nom_mois$="Janvier"
650 =2: nom_mois$="Février"
660 =3: nom_mois$="Mars"
670 =4: nom_mois$="Avril"
680 =5: nom_mois$="Mai"
690 =6: nom_mois$="Juin"
700 =7: nom_mois$="Juillet"
710 =8: nom_mois$="Août"
720 =9: nom_mois$="Septembre"
730 =10: nom_mois$="Octobre"
740 =11: nom_mois$="Novembre"
750 =12: nom_mois$="Décembre"
760 END SElect
770 END DEfine quel_mois
780 DEfine PROCEDURE graph
790 FOR n= 4 TO 7:SCALE#n, 100,0,0
800 FOR n=4 TO 7: LINE#n,15,12 TO 15,98
810 FOR n=4 TO 7: LINE#n,15,55 TO 153,55
820 FOR n=4 TO 7: AT#n,0,0:PRINT#n,"+1": AT#n,4,0:
PRINT#n," 0": AT#n,8,0:PRINT#n,"-1"
830 FOR n=4 TO 7
840 FOR j=5 TO jour STEP 5:AT#n,5,j+2: PRINT#n,j:
AT#n,0,j+2: PRINT#n,"!": AT#n,8,j+2:PRINT#n,"!"
850 END FOR n
860 quel_mois (mc)
870 AT#4,9,0: PRINT#4,"----Physique. ";nom_mois$;" ";ac:
AT#5,9,0: PRINT#5,"----Emotion. ";nom_mois$;" ";ac
880 AT#6,9,0: PRINT#6,"----Intellect. ";nom_mois$;" ";ac:
AT#7,9,0: PRINT#7,"----Intuition. ";nom_mois$;" ";ac
890 CLS#0: PRINT#0,"Biorythme de ";nom_b$: quel_mois (mn):
PRINT#0,"Date de naissance: ";jn;" ";nom_mois$;" ";an;
900 END DEfine
910 DEfine PROCEDURE courbe
920 graph
930 canal=4: REMark physique
940 y=factc-factn
950 FOR x=15 TO si

```

## BIORHYTHMES\_BAS par JC Touzin page III

```
950 FOR x=15 TO si
960 POINT#canal, x,55+43*SIN(RAD(360)*(y/s))
970 y=y+.225806
980 END FOR x
990 canal=5: REMark émotion
1000 y=factc-factn
1010 FOR x=15 TO si
1020 POINT#canal, x,55+43*SIN(RAD(360)*(y/t))
1030 y=y+.225806
1040 END FOR x
1050 canal=6: REMark intellect
1060 y=factc-factn
1070 FOR x= 15 TO si
1080 POINT#canal, x,55+43*SIN(RAD(360)*(y/z))
1090 y=y+.225806
1100 END FOR x
1110 canal=7: REMark intuition
1120 y=factc-factn
1130 FOR x=15 TO si
1140 POINT#canal, x,55+43*SIN(RAD(360)*(y/ii))
1150 y=y+.225806
1160 END FOR x
1170 IF i$=="o" : imprimante
1180 END DEFine courbe
1190 DEFine PROCedure intro
1200 WINDOW 512,256,0,0:PAPER 0:CLS
1210 fenetre 4,238,102,16,0,0
1220 fenetre 5,238,102,258,0,0
1230 fenetre 6,238,102,16,106,0
1240 fenetre 7,238,102,258,106,0
1250 fenetre 0,480,22,16,210,0
1260 END DEFine intro
1270 DEFine PROCedure fenetre (canal,x,y,ox,oy,couleur)
1280 WINDOW#canal,x,y,ox,oy
1290 PAPER#canal,couleur
1300 CLS#canal
1310 INK#canal,7*(couleur<4)
1320 BORDER#canal,1,7
1330 END DEFine fenetre
1340 DEFine PROCedure imprimante
1350 OPEN#3,ser
1360 PRINT#3,CHR$(27);"0" :REMark 1/B interligne
1370 CLOSE#3
1380 CALL a
1390 END DEFine imprimante
```

# LOTTERIE

LOTTERIE\_BAS est petit programme assez simple qui génère des combinaisons de chiffres au hasard qui peuvent être utilisés pour les loteries 6/49 et SELECT 42.

Le cœur du programme est la procédure LOTTO qui demande 2 paramètres. Le premier spécifie le nombre maximum qui peut être accepté. Le deuxième donne le nombre de chiffre à générer. Ainsi pour la 6/49, nous avons LOTTO 49,6 et pour la SELECT 42, nous avons LOTTO 42,6. On pourrait obtenir des combinaisons pour la QUOTIDIENNE 3 chiffres avec LOTTO 9,3.

Les combinaisons sont trouvées avec la procédure TROUVE\_UNE\_COMBINE qui demande 3 paramètres. Le numéro de la combinaison, le nombre maximum et nombre d'éléments composant la combinaison. Chaque élément est obtenu avec la fonction RND. On vérifie que l'élément obtenu est unique dans la combinaison. Une fois tous les éléments trouvés, on les classe par ordre croissant avec la procédure BUBBLE\_SORT.

La loterie et la probabilité		
Sélection gagnante	Chances de gagner	Gros lot minimum
<b>LOTTO 6/49</b>		
3/6 (10\$)	1 sur 56	1 000 000
4/6	1 sur 1 032	
5/6	1 sur 55 491	
5/6+	1 sur 2 330 636	
6/6	1 sur 13 983 816	
<b>SELECT 42</b>		
3/6 (5\$)	1 sur 37	500 000
4/6	1 sur 555	
5/6	1 sur 24 900	
5/6+	1 sur 874 298	
6/6	1 sur 5 245 876	

SOURCE LOTO-QUEBEC

# Jeu d'hasard: Jeu d'argent

Un sondage réalisé en 1988 par LOTO-QUEBEC mentionne que 95% des Québécois ont joué au moins une fois à la loterie dans leur vie. En 1987-88, Loto-Québec atteignait le milliard de \$ comme chiffre d'affaires. Cela signifie que chaque Québécois dépense \$150 par année en billets de loto...

En 1920, le gouvernement Taschereau tente de mettre sur pied une loterie pour financer le système d'éducation du Québec. Duplessis, dans les années 40, tente de financer les hôpitaux de cette façon. Ces deux tentatives aboutissent à des échecs à cause de pressions exercées par le gouvernement fédéral pour qui les loteries étaient immorales. C'est en 1970 que

Loto-Québec sera créé. En constatant le succès obtenu au Québec, les autres provinces emboîtent le pas dans la création de loteries provinciales et inter-provinciales (comme la 6/49).

Loto-Québec obtient 60% de son chiffre d'affaires à partir de la Loto 6/49. Sur le milliard de \$ de vente annuelle, 380 millions va au gouvernement, 477 millions aux gagnants, 54 millions est distribué aux marchands-détaillants et le reste sert à couvrir les frais d'administration.

LOTERIE\_bas par Réal Gagnon Montréal (Québec)

```
100 REMark LOTERIE bas .5
110 REMark par Réal Gagnon 1989 pour QL_DOC #11
120 :
130 MODE 4
140 WINDOW 512,256,0,0:PAPER 0:INK 7
150 BORDER 20,56
160 REPEAT boucle_principale
170   CLS
180   INK 2:PRINT " Loterie version 0.5",,:INK 7
190   INK 4:PRINT " écrit par Réal Gagnon, Mt1 Qc 1989\\":INK 7
200   PRINT " Sélection des nombres pour quelle loterie?"
210 PRINT " [ 0 ] Quitte" [ 1 ] Loto 6/49" [ 2 ] Sélect 42"
220   DIM numero(9,7),chiffre2$(7)
230   touche=CODE(INKEY$(-1))
240   SELECT ON touche
250     =CODE('1'): lotto 49,6
260     =CODE('2'): lotto 42,6
270     =CODE('0'): INK 2: PRINT "\\ " OK, à la prochaine..."
280                   INK 7: EXIT boucle_principale
290   =REMAINDER :BEEP 1,1
300   END SELECT
310 END REPEAT boucle_principale
320 :
```



```

330 DEFine PROCEDURE lotto (limite,combien)
340 LOCAL i,boucle
350 REPEAT boucle
360 PRINT "\\:CSIZE 2,1:PRINT "Combinaison pour ";
370 INK 32
380 IF limite=49
390 PRINT "LOTTO 6/49"
400 ELSE
410 PRINT "SELECT 42"
420 END IF
430 INK 7:CSIZE 0,0:PRINT "## Je pense ";
440 FOR i=1 TO 9:trouve_une_combine i,limite,combien:PRINT ". ";
450 PRINT "\\
460 FOR i=1 TO 9:CSIZE 2,0:INK 2:PRINT "##";i::CSIZE 0,0:PRINT ' ',
470 INK 7:PRINT
480 FOR i=1 TO combien
490 PRINT " ";
500 FOR ii=1 TO 8:PRINT numero(ii,i),
510 PRINT numero(9,i)
520 END FOR i
530 INK 4:PRINT "\\ " Encore (o/n) ?":INK 7
540 touche%=INKEY$(-1)
550 IF touche%=='n':EXIT boucle
560 END REPEAT boucle
570 END DEFine
580 :
590 DEFine PROCEDURE trouve_une_combine (laquelle,limite,nbr_chiffre)
600 LOCAL tampon$,i,ii
610 chiffre%=FILL$(CHR$(0),7)
620 FOR i=1 TO nbr_chiffre
630 REPEAT cherche
640 tampon%=CHR$(INT(RND(1 TO limite)))
650 IF NOT (tampon% INSTR chiffre%)
660 chiffre$(i)=tampon%:chiffre2$(i)=tampon%:EXIT cherche
670 END IF
680 END REPEAT cherche
690 END FOR i
700 bubble sort nbr_chiffre
710 FOR i=1 TO nbr_chiffre:numero(laquelle,i)=CODE(chiffre2$(i))
720 END DEFine
730 :
740 DEFine PROCEDURE bubble sort (nbr_element)
750 LOCAL a, b, tampon$, flag
760 FOR a=1 TO nbr_element
770 flag=1
780 FOR b=1 TO nbr_element-a
790 IF CODE(chiffre2$(b))>CODE(chiffre2$(b+1))
800 flag=0
810 tampon%=chiffre2$(b)
820 chiffre2$(b)=chiffre2$(b+1)
830 chiffre2$(b+1)=tampon%
840 END IF
850 END FOR b
860 IF flag THEN RETURN
870 END FOR a
880 END DEFine bubble

```

# DLO\_BAS

Version 0.6  
par Réal Gagnon

Voici une nouvelle version du programme DLO\_BAS qui nous permet de voir plus clairement le contenu d'une disquette ou d'un  $\mu$ drive. La plupart des "BUGS" présents sur la première version ont été corrigés et de nouvelles options ont fait leurs apparitions en plus d'une nouvelle interface utilisateur.

Pour débiter DLO\_BAS, on fait simplement LRUN xxxx\_DLO\_BAS où xxxx est le support contenant DLO\_BAS.

L'écran initial nous demande de mettre une disquette formatée dans flp1\_. Si on travaille avec des  $\mu$ drives seulement (ou RAMdisk), on change à la ligne 227 la variable mast\_cat\$ pour 'mdv1\_dir1' (ou 'ram1\_dir1') au lieu de 'flp1\_dir1'.

Puis on demande le nom du "device", ce dernier peut être flp1\_, flp2\_, mdv1\_ ou mdv2\_. Nous avons la possibilité d'ajouter un titre ou un commentaire, sinon on fait simplement [ENTER]. Le directory du "device" est lu, classé par ordre alphabétique et affiché. L'affichage (et l'impression) se fait sur 6 colonnes, cela peut être modifié en ajustant la variable nbr\_col à la ligne 1650. Une fois ceci fait, nous pouvons lire un autre "device", aller au menu d'impression ou bien retourner au SuperBASIC.

Pour l'impression, le programme offre 3 options:

Très petites lettres, impression en mode SUPERSCRIPT COMPRESSE, idéale pour faire des étiquettes à coller sur la disquette elle-même.

Petites lettres, impression en mode COMPRESSE qui peut servir pour une petite feuille à coller sur la pochette de la disquette.

L'option impression normale imprime sur toute la largeur d'une feuille normale en 80 caractères par ligne en 6 colonnes.

```

100 REMark -----
110 REMark -----Disk Label Organizer v.6
115 REMark -----par Réal Gagnon Mtl Qc 89
117 REMark -----
130 WINDOW SIZE,256,0,0
140 BORDER 3,45:PAPER 0:INK 2:CLS
170 PRINT 'Diskette Label Organizer',
    'par Réal Gagnon, Mtl Québec 1989'
180 PRINT 'version 0.6',,,
    'Logiciel du domaine public pour le Sinclair QL'
190 INK 7
200 PRINT "IMPORTANT: Disquette NON-PROTEGEE en
    écriture dans FLPI..."\
210 INPUT 'Contenu de quel "device" (ex. flpl_) ? ';device$
220 INPUT 'Avez un TITRE ou une NOTE spéciale,
    sinon faites [ENTER] ? ';titre$
225 :
227 mast_cat$='flpl_dir1'
230 read_dir: IF empty=0 : RUN
240 sort_fi$
250 list_scr
260 :
270 BORDER#0,1,7:CLS#0
290 PRINT#0, '[1] Lire un autre "device",, '[2] Imprime le contenu'
291 PRINT#0, '[0] Retour au SuperBASIC"
300 REPEAT loop
310 a=CODE(INKEY$(-1))
320 SELEct ON a
330 =CODE("1"):RUN
340 =CODE("2"):choi prt
350 =CODE("0"):PRINT#0, "Retour au SuperBASIC...":STOP
360 END SELEct
370 END REPEAT loop
380 :
390 DEFine PROCedure choi_prt
400 CLS#0
410 REPEAT attend:IF INKEY$="":EXIT attend:ELSE :END REPEAT attend
420 PRINT#0, '[1] Petites lettres",, "[2] Très petites lettres"
421 PRINT#0, "[3] Impression normale",, "[0] Menu principal"
425 REPEAT loop2
430 a=CODE(INKEY$(-1))
440 SELEct ON a
450 =CODE('1'):list_prt 1
460 =CODE('2'):list_prt 2
462 =CODE('3'):list_prt 3
465 =CODE('0'):EXIT loop2
470 END SELEct
475 END REPEAT loop2
480 CLS#0
490 PRINT#0, '[1] Lire un autre "device",, '[2] Imprime le contenu'

```

```

491 PRINT#0,"[0] Retour au SuperBASIC"
500 END DEFine
510 :
520 DEFine PROCedure read_dir
530 empty=1
540 DELETE mast_cat$
550 OPEN NEW#5,mast_cat$
560 DIR#5,device$
570 CLOSE#5:OPEN#5,mast_cat$
590 REMark -----recherche le nombre de secteurs libres
610 CLS
620 PRINT "Lecture/Analyse du contenu de ";device$,
630 IF EOF(#5)
631 PRINT \'Il n'y a aucun fichier sur ";device$
632 PRINT "Appuyez une touche...": PAUSE: CLOSE#5:empty=0:RETURN
633 PAUSE: CLOSE#5:empty=0:RETURN
636 END IF
640 INPUT#5,fr_sp$
650 PRINT fr_sp$
670 REMark -----trouve combien il y a de fichier et
671 REMark -----lequel est le plus long
690 INPUT#5,dummy$ :REMark ligne vide
700 nfiles=0
710 lmax=0
720 REPEAT loop
730 INPUT #5,a$
750 lfi=LEN(a$)
760 IF lfi>lmax:lmax=lfi
770 IF a$<>'dir1':nfiles=nfiles+1
775 IF EOF(#5):EXIT loop
780 END REPEAT loop
790 PRINT 'Il y a ',nfiles,' fichiers.'
800 CLOSE#5:OPEN#5,mast_cat$
810 INPUT#5,dummy$:INPUT#5,dummy$ :REMark secteurs libres+ligne vide
830 REMark -----creation du tableau contenant les noms
831 REMark -----de fichier.
850 DIM fi$(nfiles,lmax)
890 i=1
900 REPEAT loop
910 INPUT#5,a$
930 IF a$<>'dir1'
940 fi$(i)=a$
950 i=i+1
960 END IF
965 IF EOF(#5):EXIT loop:END IF
970 END REPEAT loop
980 CLOSE#5: DELETE mast_cat$
990 END DEFine
1010 REMark -----PROCs sort & qsort
1011 REMark -----source SINCLAIR QL WORLD!

```

```

1020 :
1030 DEFine PROCedure sort (array)
1040 INK 4
1050 PRINT 'Je classe ...'
1060 INK 7
1070 qsort array,1,DIMN(array)
1080 END DEFine sort
1090 :
1100 DEFine PROCedure qsort (array,bottom,top)
1110 LOCAL loop,lo,hi,pointer
1120 lo=bottom:hi=top
1130 pointer=bottom
1140 REPEAT loop
1150     IF lo)=hi:EXIT loop
1160     IF array(lo)>array(hi)
1170         temp=array(lo)
1180         array(lo)=array(hi)
1190         array(hi)=temp
1200         IF pointer=lo
1210             lo=lo+1:pointer=hi
1220         ELSE
1230             hi=hi-1:pointer=lo
1240         END IF
1250     ELSE
1260         IF pointer=lo
1270             hi=hi-1
1280         ELSE
1290             lo=lo+1
1300         END IF
1310     END IF
1320 END REPEAT loop
1330 IF ABS(top-bottom)<2 THEN RETURN
1340 qsort array, bottom, pointer-1
1350 qsort array, pointer+1, top
1360 END DEFine qsort
1370 :
1390 DEFine PROCedure list_prt (style)
1395 esc%=CHR$(27)
1400 l$='
1410 OPEN#5,ser1
1430 st=style
1435 SElect ON st
1437     =1
1460     PRINT#5,esc%;"M";CHR$(15);esc%;"SO";esc%;"3";CHR$(16);
1470     =2
1481     PRINT#5,esc%;"M";CHR$(15);
1485     =3
1487     PRINT#5,esc%;"P";
1488 END SElect
1510 PRINT#5,esc%;"W1";titre$\fr_sp%;esc%;"W0":REMark large on/off

```

```

1520 display_files
1530 PRINT#5,esc$;"e"////////////////////////////////://CLOSE#5
1540 END DEFine
1550 :
1560 DEFine PROCEDURE list_scr
1570 l$='
1580 OPEN#5,scr 512x180a0x33:BORDER#5,2,2:PAPER#5,0:INK#5,4:CLS#5
1590 PRINT#5,tiÈre$\\vr_sp$
1600 display_files
1610 END DEFine
1620 :
1630 DEFine PROCEDURE display_files
1640 REMark -----NBR_COL=nombre de colonne
1650 nbr_col=6
1660 n_lignes=INT(nfiles/nbr_col)
1670 FOR ligne=0 TO n_lignes
1680   FOR colonne=1 TO nbr_col
1690     IF ((ligne#6)+colonne)>nfiles:EXIT ligne
1700     long_f=LEN(fi$((ligne#6)+colonne))
1710     IF long_f>12
1720       f$=fi$((ligne#6)+colonne),1 TO 11)
1730     ELSE
1740       f$=fi$((ligne#6)+colonne)
1750     END IF
1760     l f=13-LEN(f$)
1770     PRINT #5,f$;l$(1 TO l_f);
1780   END FOR colonne
1790   PRINT#5
1800 END FOR ligne
1810 END DEFine

```

Lecture/Analyse du contenu de filp1\_

Il y a 67 fichiers.

Je classe ...

Disquette #2 PAGE DESIGNER

ALIEN_font	AMELIA_font	ARCADE_font	AURA_font	BABYTEETH_f	BOLDSTANDAR
BOLD_font	BOTTLENECK_	BOXES_patn	BROADWAY_EN	COUNTDOWN_f	COUNTDOWN_h
CUBIC_hires	DATA70_font	DATA70_hires	DIAMONDS_pa	DOUBLEOUTLI	DOWNWARD_fo
EMPTY#4_page	EMPTY#Q_page	EMPTY_LASER	EMPTY_SCRREE	EXAMPLE_cut	EXAMPLE_page
EXAMPLE_scr	EXTRABOLD_f	FANCY_font	FILLS1_patn	FILLS2_patn	FILLS3_patn
FONT#_page	FRAME_patn	FUTURABLACK	FUTURADISPL	FUTURADISPL	HALFCHECK_p
HALFHEIGHT_	HEAVY_font	ITALIC_font	MOORECOMPUT	ODIN_font	OLDE_font
OUTLINE_font	OUTLINE_hir	OUTLINE_patn	OVAl_patn	PROMOTION_p	REVERSED_fo
SANSERIF_f	SERIFA_hires	SERIF_font	SMALL_font	SPOOLER_task	SQUARESBOL
SQUARESS_fo	SQUARE_font	STANDARD_fo	STANDARD_hi	STENCIL_font	TINES_hires
TRAFFIC_font	TRIANGLES_p	UPSIDEDOWN_	UPWARD_font	WAVES_patn	WIDE_font
ZIPPER_font					

[1] Lire un autre "device"

[2] Imprime le contenu

[0] Retour au SuperBASIC

# Trucs QLIens Astuces QLIennes

La date de fabrication d'un µdrive est indiquée sur celui-ci sous la forme d'un code composé de 4 chiffres, par exemple "2315".

Les 3 premiers chiffres représente la journée de fabrication, par exemple "231" serait le 231e jour de l'année qui est représentée par le dernier chiffre. Si ce dernier est "5" alors c'est de l'année "1985" qu'il s'agit.

Le code "1416", par exemple, indique que la cartouche a été fabriquée la 141e journée de l'année 1986.

---

Il est possible de transformer une imprimante SEIKOSHA SP1000 en une "SINCLAIR QL PRINTER" en installant une copie de la ROM de cette dernière dans la SP1000. Cette ROM est contenue dans un EPROM 27128. Il faudra mettre tous les interrupteurs de la SEIKOSHA à "OFF" pour fonctionner en mode QL.

## QL DOC

Le Bulletin Québécois  
pour Sinclair QL

Known variously as Seattle Computer 86-DOS, IBM Personal Computer DOS, and Zenith Z-DOS, MS-DOS was developed by Seattle Computer Products for its 8086-based computer system. The MS-DOS history is intertwined with the general development of software for 8086-based computers.

In May 1979, Seattle Computer made the first prototype of its 8086 microprocessor card for the S-100 bus. There were brief discussions with Digital Research about using one of Seattle Computer's prototypes to aid in developing CP/M-86, which was to be ready "soon." Although Seattle Computer was considering using CP/M-86 when it became available (expected no later than the end of 1979), there were only two working prototypes of the 8086 processor card, and it was felt that both were needed in house. Therefore, there wasn't one free for Digital Research.

Microsoft had already started a strong 8086 software-development program. The firm was ready to try the 8086 version of Stand-Alone Disk BASIC, which is a version of its BASIC interpreter with a built-in operating system. During the last two weeks of May 1979, this BASIC was made completely functional using the hardware

last less than six months. (Unfortunately, it has lasted much longer than that as part MS-DOS.)

In the last few days of 1980, a new version of the DOS was released, now known as 86-DOS version 0.3. Seattle Computer passed this new version on to Microsoft, which had bought non-exclusive rights to market 86-DOS and had one customer for it at the time. Also about this time, Digital Research released the first copies of CP/M-86. In April 1981, Seattle Computer Products released 86-DOS version 1.00, which was very similar to the versions of MS-DOS that are widely distributed today.

## A Short History of MS-DOS

that Seattle Computer provided for Microsoft. Seattle Computer Products displayed the complete package (8086 running disk BASIC) in New York the first week of June at the 1979 National Computer Conference. (This was the first-ever public display of an 8086 BASIC and of an 8086 processor card for the S-100 bus.)

Seattle Computer shipped its first 8086 cards in November 1979, with Stand-Alone Disk BASIC as the only software to run on it. The months rolled by, and CP/M-86 was nowhere in sight. Finally, in April 1980, Seattle decided to create its own DOS. This decision resulted just as much from concern about CP/M's shortcomings as from the urgent need for a general-purpose operating system.

The first versions of the operating system, called QDOS 0.10, were shipped in August 1980. QDOS stood for Quick and Dirty Operating System because it was thrown together in such a hurry (two man-months), but it worked surprisingly well. It had all the basic utilities for assembly-language development except an editor. One week later, Seattle Computer had created an operating system with an editor, an absurdity known as EDLIN (editor of lines). A primitive line-oriented system, it was supposed to

In July 1981, Microsoft bought all rights to the DOS from Seattle Computer, and the name MS-DOS was adopted. Shortly afterward, IBM announced the Personal Computer, using as its operating system what was essentially Seattle Computer's 86-DOS 1.14. Microsoft has been continuously improving the DOS, providing version 1.24 to IBM (as IBM's version 1.1) with MS-DOS version 1.25 as the general release to all MS-DOS customers in March 1982. Now version 2.0, released in February 1983, has just been announced with IBM's new XT computer.